

**ESD RECORD COPY**

RETURN TO  
SCIENTIFIC & TECHNICAL INFORMATION DIVISION  
(ESTI), BUILDING 1211

**ESD ACCESSION LIST**

ESTI Call No. 63608  
Copy No. 1 of 1 cys.

*ESLE***Technical Note****1968-24**

**Numerical Simulation  
of the Attitude Equations  
of a Satellite Containing  
Damping Rotors  
and a Reaction Wheel**

**R. W. Brockett  
N. M. Brody**

**15 November 1968**

Prepared under Electronic Systems Division Contract AF 19(628)-5167 by

**Lincoln Laboratory****MASSACHUSETTS INSTITUTE OF TECHNOLOGY****Lexington, Massachusetts***AD679992*





MASSACHUSETTS INSTITUTE OF TECHNOLOGY  
LINCOLN LABORATORY

NUMERICAL SIMULATION  
OF THE ATTITUDE EQUATIONS OF A SATELLITE  
CONTAINING DAMPING ROTORS AND A REACTION WHEEL

*R. W. BROCKETT*

*NAOMI M. BRODY*

*Group 63*

TECHNICAL NOTE 1968-24

15 NOVEMBER 1968

LEXINGTON

MASSACHUSETTS

## ABSTRACT

This report describes a program for simulating the motion of a rigid body containing a reaction wheel, damping media and torquing (magnetic or otherwise). The dynamical equations are simulated in terms of the angular velocities in body-fixed coordinates and the kinematic equations are simulated in terms of direction cosines. The entire system of equations has been programmed using a fourth order predictor-corrector method with automatic step-size adjustment and a fourth order Runge-Kutta starting routine. The ratio of computer time to real time is between 1 to 10 and 2 to 1 for LES-5-type satellites. The potential user need only be familiar enough with this report to fill out the input data sheets on pages vii - ix.

Accepted for the Air Force  
Franklin C. Hudson  
Chief, Lincoln Laboratory Office

## CONTENTS

	Abstract	iii
	Input Data Sheet	vii
I.	Introduction	1
II.	Dynamical Models And Their Equations of Motion	2
III.	Kinematics	6
IV.	External Torques	10
V.	Numerical Procedures and Results	10
VI.	Sample Problem 1: Torque-Free Motion	12
VII.	Sample Problem 2: A Magnetic Attitude Control Law	17
VIII.	Sample Problem 3: Reaction Wheel Spin Down	23
	Appendix 1: Usage of the Attitude Stabilization Program	28
	References	40

## INPUT DATA SHEET

### I. Kinematic Specifications

Fill in either A or B

A. Initial values of direction cosines:

$$\underline{A}(0) = \begin{pmatrix} \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \\ \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \\ \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \end{pmatrix}$$

B. Initial Euler angles:

$$\theta_0 = \underline{\hspace{2cm}} \text{ rad}, \varphi_0 = \underline{\hspace{2cm}} \text{ rad}, \psi_0 = \underline{\hspace{2cm}} \text{ rad}$$

### II. Dynamic Specifications

The  $i^{\text{th}}$  moment of inertia and the  $i^{\text{th}}$  component of angular velocity are assumed to be taken about the  $i^{\text{th}}$  principal axis of the satellite. If there is a reaction wheel, the axes must be numbered so as to place the wheel on the 3<sup>rd</sup> axis.

A. Satellite Characteristics

Initial value of angular velocity:

$$\underline{\omega}(0) = \begin{pmatrix} \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}}, & \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}}, & \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}} \end{pmatrix}$$

Moments of inertia:

$$I_1 = \underline{\hspace{2cm}} \text{ kg m}^2, I_2 = \underline{\hspace{2cm}} \text{ kg m}^2, I_3 = \underline{\hspace{2cm}} \text{ kg m}^2$$

B. Rotor Characteristics

If there is a spherical damping rotor on the  $i^{\text{th}}$  axis, fill in  $\chi_i(0)$ , its initial angular velocity;  $R_i$ , its moment of inertia; and  $k_i$ , its damping constant. If no rotor, write NONE in the corresponding blanks.

$$\begin{aligned} \underline{\chi}(0) &= \begin{pmatrix} \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}}, & \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}}, & \underline{\hspace{2cm}} \frac{\text{rad}}{\text{sec}} \end{pmatrix} \\ R_1 &= \underline{\hspace{2cm}} \text{ kg m}^2, R_2 = \underline{\hspace{2cm}} \text{ kg m}^2, R_3 = \underline{\hspace{2cm}} \text{ kg m}^2 \\ k_1 &= \underline{\hspace{2cm}} \frac{\text{kg m}^2}{\text{sec}}, k_2 = \underline{\hspace{2cm}} \frac{\text{kg m}^2}{\text{sec}}, k_3 = \underline{\hspace{2cm}} \frac{\text{kg m}^2}{\text{sec}} \end{aligned}$$

### C. Reaction Wheel Characteristics

If the satellite has a reaction wheel, fill in  $\chi_3$ , its initial angular velocity, and  $W_1$ ,  $W_2$ ,  $W_3$ , its moments of inertia. Also supply a control law determining  $\ell_3$  ( $A$ ,  $\omega$ ,  $\chi_3$ ), the torque of the reaction wheel on the satellite. If no wheel, write NONE.

$$\begin{array}{rcl} \chi_3 & = & \frac{\text{rad}}{\text{sec}} \\ W_1 = W_2 & = & \text{kg m}^2 \quad W_3 = \text{kg m}^2 \\ \ell_3 & = & \end{array}$$

### III. External Torque

Supply a control law determining  $\underline{n}$ , the external torque on the satellite, in body-fixed coordinates.

$$\begin{array}{rcl} n_1 & = & \text{nt m} \\ n_2 & = & \text{nt m} \\ n_3 & = & \text{nt m} \end{array}$$

### IV. Termination Criterion

Fill in A or B, or both. If both, the first criterion to be met will terminate the simulation.

- A. Terminate after integrating over a period of \_\_\_\_\_ sec.
- B. Run the simulation until the solutions satisfy the following condition:

---

---

---

### V. Title for Output

Give a brief title for the output, 128 characters or less; a character is a letter, a numeral, a punctuation mark, or a space.

---

---



# NUMERICAL SIMULATION OF THE ATTITUDE EQUATIONS OF A SATELLITE CONTAINING DAMPING ROTORS AND A REACTION WHEEL

## I. INTRODUCTION

Our objective has been to develop a computer program for solving the equations which describe the motion of a satellite about its center of mass and to get a feeling for how accurate the simulation might be. The major decisions which needed to be made were:

- (i) What is the simplest model of the satellite which can account for the main features of the motion. In particular, how can we include damping of wobble without introducing too much complexity?
- (ii) Which coordinates should be chosen in carrying out the simulation-- Euler angles, quaternions, Cayley-Klein parameters, direction cosines or some other set.
- (iii) What method of integration should be chosen, e.g. Runge-Kutta, Adams-Bashforth, etc., and what step sizes and error bounds (in automatic step size control) should be used.

Naturally the decision in each case was made in terms of what we felt would maximize ultimate usefulness of the program. The uses we had in mind are exemplified by the following, all of which have been suggested to us by other people in the Laboratory working on LES-5,-6 and-7.

- (i) Suppose a rigid body has within it a motor driven reaction wheel. The power fails and the wheel spins down because of friction. How does this affect the motion of the main body over a period of time long enough to let the wheel lose 90% of its relative angular velocity?
- (ii) Suppose a rigid body has magnetic rods and a control law for orienting it along a magnetic field. How large can the magnetic torques be without causing instability and how fast can such a system be made to respond?
- (iii) Given a particular configuration of gas jets and a particular control law, how much fuel will it take to orient a satellite from a given initial state?

This report does not contain any true applications in the sense of detailed parametric studies of a particular problem. Sample problems however have been worked and the results are given along with the computation times and some remarks on accuracy.

From the point of view of the potential engineering user the main fact is that by filling out the input data sheet (pages vii - ix) and handing it to a



programmer he can in a short time get a complete simulation without ever studying the internal workings of the program. In order to assure useability independent of programming personnel we have included a reasonably comprehensive guide to the program in an appendix.

## II. DYNAMICAL MODELS AND THEIR EQUATIONS OF MOTION

The actual mechanisms which account for wobble damping in most cases are far too complex to simulate in detail. Perhaps the simplest model which bears resemblance to the physical situation is that of a small rotor in an otherwise rigid body with the axis of the rotor along one of the principle axes. If the center of mass of the rotor coincides with the center of mass of main body and if the rotor is dynamically spherical then further simplifications occur. For the sake of symmetry one may also want to consider additional rotors along the other principal axes.

### A. Case 1: 3 Spherical Damping Rotors

Lets now consider the case of a rigid body with moments of inertia  $I_1$ ,  $I_2$  and  $I_3$  about its principal axes. We further suppose that the body contains spherical rotors having as their moment of inertia about any axis passing through their center of mass,  $R_1$ ,  $R_2$  and  $R_3$  respectively. These rotors are mounted in such a way as to permit the  $i^{\text{th}}$  rotor to rotate about the  $i^{\text{th}}$  principal axis. If we let  $\omega_1$ ,  $\omega_2$  and  $\omega_3$  denote the angular velocities of the main body and if  $\dot{\chi}_i$  denotes the angular velocity of the  $i^{\text{th}}$  rotor (see Fig. 1), then the full equations of motion are:

$$(R_2 + R_3 + I_1) \dot{\omega}_1 = (I_2 - I_3) \omega_2 \omega_3 + \ell_1 + n_1$$

$$(R_1 + R_3 + I_2) \dot{\omega}_2 = (I_3 - I_1) \omega_1 \omega_3 + \ell_2 + n_2$$

$$(R_1 + R_2 + I_3) \dot{\omega}_3 = (I_1 - I_2) \omega_1 \omega_2$$

$$R_1 \dot{\chi}_1 = -\ell_1$$

$$R_2 \dot{\chi}_2 = -\ell_2$$

$$R_3 \dot{\chi}_3 = -\ell_3$$

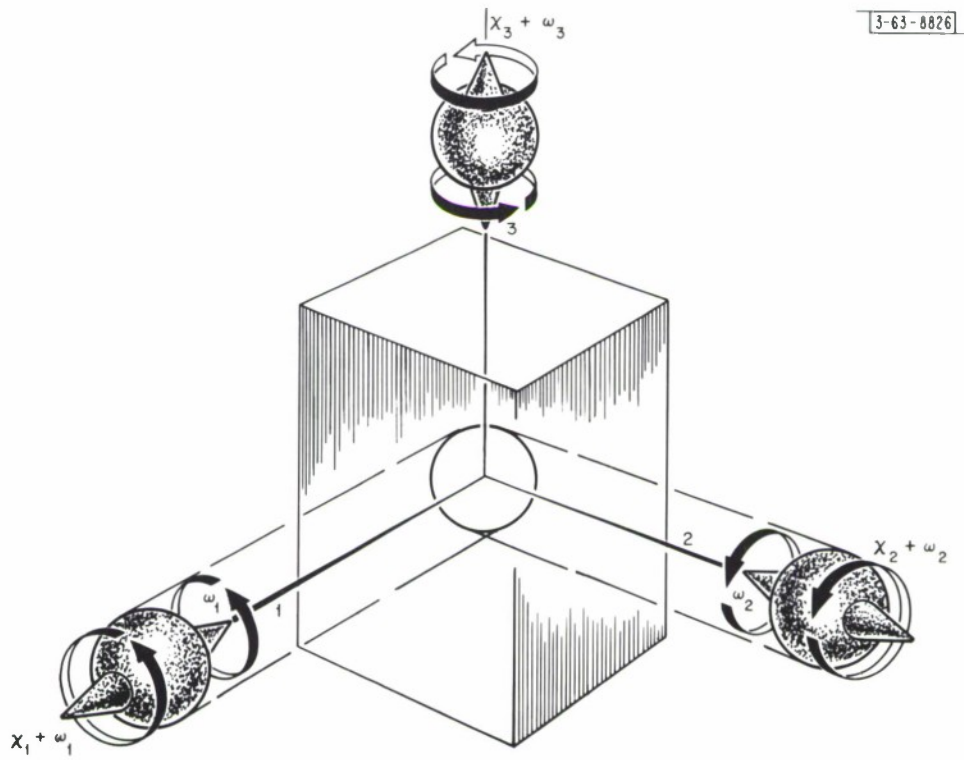


Fig. 1. A rigid body with three pivoted spherical rotors at the center of mass. See section 2.1, case 1, for equations of motion. Viscous frictions between rotors and main body provides damping. Rotors are shown withdrawn for ease of visualization.

where  $\ell_i$  is the component of the reaction torque between the  $i^{\text{th}}$  wheel and the main body lying along its axis of rotation. To get damping we can assume viscous friction in the bearing supporting the reaction wheels. If this is linear then

$$\ell_i = k_i (\chi_i - \omega_i) \quad ; \quad k_i \geq 0$$

and the above equations become

$$(R_2 + R_3 + I_1) \dot{\omega}_1 = (I_2 - I_3) \omega_2 \omega_3 + k_1 (\chi_1 - \omega_1) + n_1$$

$$(R_1 + R_3 + I_2) \dot{\omega}_2 = (I_3 - I_1) \omega_1 \omega_3 + k_2 (\chi_2 - \omega_2) + n_2$$

$$(R_1 + R_2 + I_3) \dot{\omega}_3 = (I_1 - I_2) \omega_1 \omega_2 + k_3 (\chi_3 - \omega_3) + n_3$$

$$R_1 \dot{\chi}_1 = k_1 (\omega_1 - \chi_1)$$

$$R_2 \dot{\chi}_2 = k_2 (\omega_2 - \chi_2)$$

$$R_3 \dot{\chi}_3 = k_3 (\omega_3 - \chi_3)$$

#### B. Case 2: A Reaction Wheel and 2 Spherical Damping Rotors

We now consider the possibility of a reaction wheel along one of the principal axes, say the 3-axis. To keep the notation simple we will suppose that there is no damping rotor about this axis but that spherical damping rotors may be present along the other two principal axes. It does not seem to be general enough to require the reaction wheel to be dynamically spherical so we let it be cylindrical with moments of inertia  $W_1$ ,  $W_2$  and  $W_3$  (see Fig. 2). In the program, the coordinate axes are always in such a position that  $W_1 = W_2$ . In the interest of simplicity we assume that the center of mass of the reaction wheel coincides with the center of mass of the main body. In this case the equations are

$$(I_1 + R_2 + W_3) \dot{\omega}_1 = (I_2 - I_3) \omega_2 \omega_3 + (W_2 - W_3) \omega_2 \chi_3 + \ell_1 + n_1$$

$$(I_2 + R_1 + W_3) \dot{\omega}_2 = (I_3 - I_1) \omega_1 \omega_3 + (W_3 - W_1) \omega_1 \chi_3 + \ell_2 + n_2$$

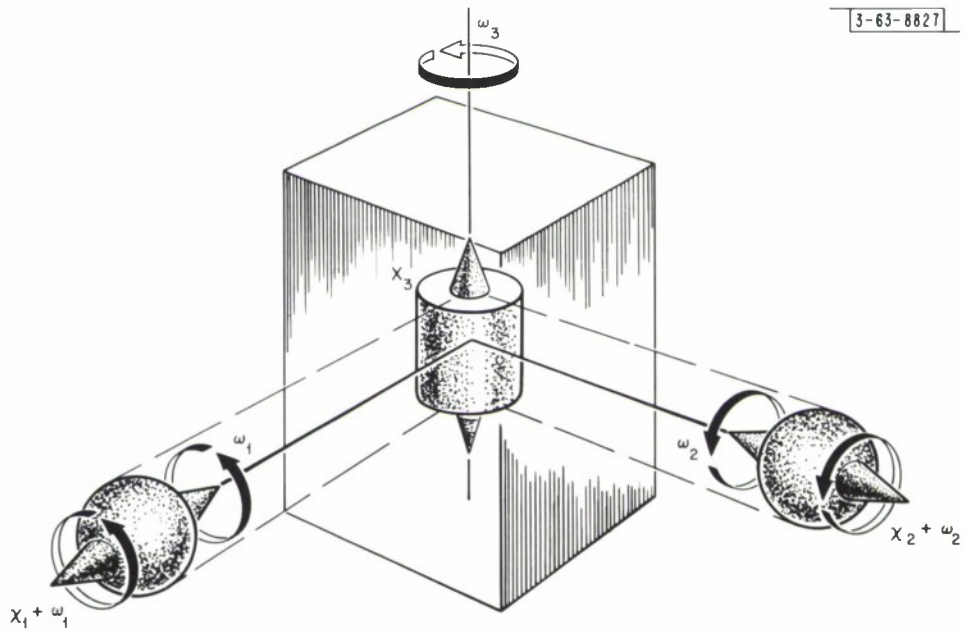


Fig. 2. Figure 1 with the rotor about the 3-axis replaced by a cylindrical reaction wheel. See section 2.2. The two rotors are spherical.



$$(I_3 + R_1 + R_2) \dot{\omega}_3 = (I_1 - I_2) \omega_1 \omega_2 + \ell_3 + n_3$$

$$R_1 \dot{\chi}_1 = -\ell_1$$

$$R_2 \dot{\chi}_2 = -\ell_2$$

$$W_3 \dot{\chi}_3 = -\ell_3$$

As before,  $\ell_i$  is that component of the reaction torque of the  $i^{\text{th}}$  rotor on the main body which lies along the axis of rotation of the  $i^{\text{th}}$  rotor.

Typically we let  $\ell_1$  and  $\ell_2$  be

$$\ell_1 = R_1 (\chi_1 - \omega_1)$$

$$\ell_2 = R_2 (\chi_2 - \omega_2)$$

The choice of  $\ell_3$ , like the choice of  $n_1$ ,  $n_2$  and  $n_3$  must be made so as to achieve the desired attitude.

### III. KINEMATICS

One has several choices when it comes to describing the orientation of a rigid body with respect to an inertially fixed reference frame. As is well known, the popular Euler angles are not well suited for numerical work because of singular points. The quaternions are superior in this respect but because of our need for direction cosine information in computing magnetic torques we have chosen to use direction cosines throughout. This necessitates integrating 9 kinematic equations instead of 4, as would be the case for the quaternions, but these equations are well behaved and for problems which would require the calculation of direction cosines at any stage this seems like an appropriate choice. (We have thus far not compared integration times with an integration procedure based on quaternions.)

Figure 3 shows a moving triad and a fixed triad having a common origin. We define  $a_{ij}$  as the orthogonal projection of a unit vector along the  $i^{\text{th}}$  axis

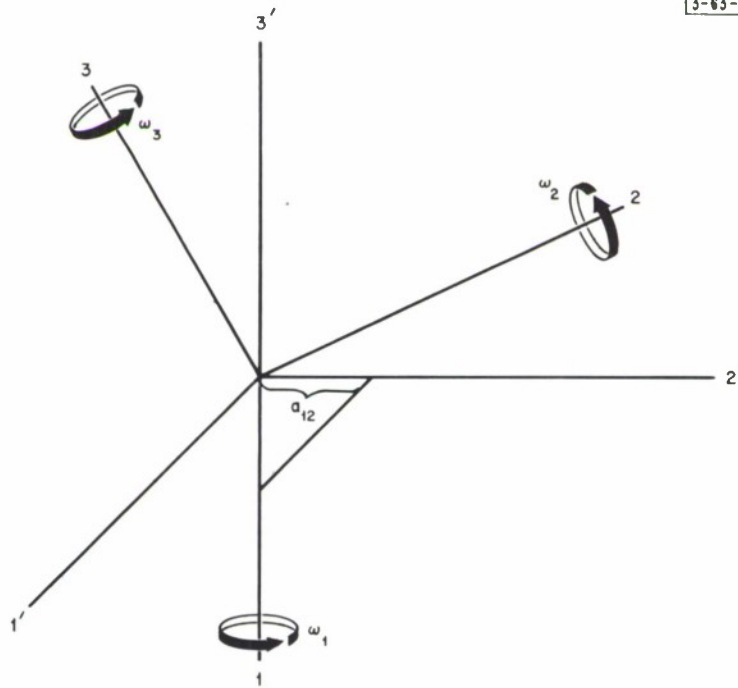


Fig. 3. The moving triad and the fixed (inertial) triad. Note definition of  $a_{12}$ ; other  $a$ 's are defined analogously.

of the moving system on the  $j^{\text{th}}$  axis of the fixed system. If we let  $\omega_i$  be the angular velocity of the moving system about its  $i^{\text{th}}$  axis then the equations of evaluation for the direction cosines are

$$\dot{a}_{11} = a_{21} \omega_3 - a_{31} \omega_2$$

$$\dot{a}_{12} = a_{22} \omega_3 - a_{32} \omega_2$$

$$\dot{a}_{13} = a_{23} \omega_3 - a_{33} \omega_2$$

$$\dot{a}_{21} = a_{31} \omega_1 - a_{11} \omega_3$$

$$\dot{a}_{22} = a_{32} \omega_1 - a_{12} \omega_3$$

$$\dot{a}_{23} = a_{33} \omega_1 - a_{13} \omega_3$$

$$\dot{a}_{31} = a_{11} \omega_2 - a_{21} \omega_1$$

$$\dot{a}_{32} = a_{12} \omega_2 - a_{22} \omega_1$$

$$\dot{a}_{33} = a_{13} \omega_2 - a_{23} \omega_1$$

Expressed more succinctly,

$$\underline{\dot{A}} = \underline{\Omega} \underline{A}$$

where

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} ; \quad \underline{\Omega} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}$$

We remark in passing that  $\underline{A}$  is an orthogonal matrix so  $\underline{A}'\underline{A}$  is the 3 by 3 identity matrix, where prime denotes transpose.

For the sake of completeness we include the formula relating the direction cosines and the Euler angles.

### A. Euler Angles in Terms of Angular Velocities and Direction Cosines

$$\theta = \arccos a_{33}$$

$$\varphi = \arccos (-a_{32}/\sin \theta)$$

$$\psi = \arccos (a_{23}/\sin \theta)$$

$$\dot{\theta} = \omega_1 \cos \psi - \omega_2 \sin \psi$$

$$\dot{\varphi} = \omega_1 \sin \psi / \sin \theta + \omega_2 \cos \psi / \sin \theta$$

$$\dot{\psi} = \omega_3 - \omega_2 \cos \psi \cos \theta / \sin \theta - \omega_1 \sin \psi \cos \theta / \sin \theta$$

### B. Directions Cosines in Terms of Euler Angles

$$\underline{A} = \begin{pmatrix} c \psi c \varphi - c \theta s \varphi s \psi & c \psi s \varphi + c \theta c \varphi s \psi & s \psi s \theta \\ -s \psi c \varphi - c \theta s \varphi c \psi & -s \psi s \varphi + c \theta c \varphi c \psi & c \psi s \theta \\ s \theta s \varphi & -s \theta c \varphi & c \theta \end{pmatrix}$$

### C. Angular Velocities in Terms of Euler Angles

$$\omega_1 = \dot{\varphi} \sin \theta \sin \psi + \dot{\theta} \cos \psi$$

$$\omega_2 = \dot{\varphi} \sin \theta \cos \psi - \dot{\theta} \sin \psi$$

$$\omega_3 = \dot{\varphi} \cos \theta + \dot{\psi}$$

In many calculations the angular momentum plays a key role. It may be shown that the matrix

$$\underline{L} = (\underline{A}' \underline{I}^{-1} \underline{\Omega} \underline{I}^{-1} \underline{A}) \det \underline{I}$$

is antisymmetric and  $l_{23}$ ,  $-l_{13}$  and  $l_{12}$  are the components of the angular momentum along the 1', 2' and 3' axes, respectively, in the inertial triad.\*

---

\*It is of some interest to note that the Euler equations for a rigid body can be written in this matrix notation as

$$(\det I) \underline{I}^{-1} \dot{\underline{\Omega}} \underline{I}^{-1} = \underline{\Omega}^2 \underline{I} - \underline{I} \underline{\Omega}^2 + \underline{N}$$



#### IV. EXTERNAL TORQUES

It is essential to keep in mind that the basic dynamical equations are written in body fixed coordinates. The torques must be expressed in body fixed coordinates and must be resolved along the principal axes of the body. Thus our equations take the simplest form if thrusters, magnetic rods, etc. are located in such a way as to make the resolution of torques along the principal axes simple. Because of the importance of magnetic torquing we will illustrate the circulation of torques using this as an example.

Consider a magnetic field of intensity  $H$  directed along the 3-axis of a coordinate system fixed in space. Suppose there is a set of 3 orthogonal magnets having moments  $m_1$ ,  $m_2$  and  $m_3$ , respectively. To calculate the torque about the  $i^{\text{th}}$  moving axis due to the  $j^{\text{th}}$  magnet we need to compute the cross product of the two vectors. Expressed in terms of the moving coordinate system,  $\underline{h}$  has the components

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ h \end{bmatrix} = h \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}$$

Hence we have

$$\begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \times \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \end{bmatrix} = \begin{bmatrix} m_2 a_{33} - m_3 a_{23} \\ m_3 a_{13} - m_1 a_{33} \\ m_1 a_{23} - m_2 a_{13} \end{bmatrix}$$

#### V. NUMERICAL PROCEDURES AND RESULTS

The attitude stabilization program employs a fourth-order predictor-corrector method of numerical integration, with the predictor calculated from the Adams-Bashforth formula, and the corrector from the Adams-Moulton formula. Because iterating the corrector formula proved very costly in execution

time and only marginally profitable in error reduction, the first approximation to the corrector formula has been taken as the final value. Starting values for the Adams-Moulton routine are obtained by fourth-order Runge-Kutta integration.

To achieve the greatest accuracy in the least time, careful control must be exercised over the integration step size. Therefore, after each step, a measure of the truncation error is evaluated as follows for each of the fifteen integral curves.

$$\eta = |(y_c - y_p)/y_c|$$

where  $y_c$  is the corrected value of  $y$ , and  $y_p$  is the predicted value. (Since the quotient may be misleading when  $y_c$  is near zero,  $\eta$  is set to zero whenever  $|y_c| \leq 5 \times 10^{-6}$ .) The step size for the next step is determined by the value of  $\eta_{\max}$ , the largest of the fifteen error estimates. Lower and upper bounds on  $\eta_{\max}$ , GLB and LUB, are supplied as input data. If  $\eta_{\max}$  is less than GLB, the step size,  $h$ , is doubled; if  $\eta_{\max}$  is greater than LUB,  $h$  is halved. When the step size is changed, the new starting values are computed by fourth-order Runge-Kutta integration.

Because the Runge-Kutta method requires twice as many derivative evaluations per step, it was expected to be about twice as slow as Adams-Moulton integration. As it turned out, an Adams-Moulton step without step size control was only 1.2 times as fast as a Runge-Kutta step, and an Adams-Moulton step with step size control took about as much computer time as Runge-Kutta step did. Although nearly as fast in our application as the Adams-Moulton method, Runge-Kutta integration is not amenable to step size control. Therefore, Runge-Kutta has been relegated to the starting routine, and Adams-Moulton integration has been chosen for the actual simulation.

Double precision (about 16 significant decimal digits) is used throughout the program, for single precision (about 8 significant digits) proved to be inadequate. After only 30 sec of simulation time, the discrepancy between the w integral curves obtained with single vs. double precision is already 0.2%

even with a step size as small as 0.01 sec. The most commonly observed mean step sizes are about ten times as big, and as step size increases, the gap between single and double precision accuracy widens exponentially.

With  $GLB = 10^{-7}$  and  $LUB = 8 \times 10^{-5}$ , the program has the following speed and accuracy. The ratio of simulation time to real time is on the order of 10:1, with the exact value determined by the complexity of the derivative functions. Over the first two hours of simulation time, the mean error in the solutions does not exceed 0.2%; over the first three hours it does not exceed 0.3%. The error may remain well below these upper limits.

## VI. SAMPLE PROBLEM 1: TORQUE-FREE MOTION

In order to fix ideas and to check the numerical procedure against a soluble problem, we consider first the case of the torque-free motion of a spinning cylindrical body without damping. The physical situation is shown in Fig. 4. The completed data sheets are on the following two pages. In order to get an analytical solution we observe that the dynamical equations are

$$\begin{aligned}\dot{\omega}_1 &= [(I - I_3)/I] \omega_2 \omega_3 \\ \dot{\omega}_2 &= [(I_3 - I)/I] \omega_1 \omega_3 \\ \dot{\omega}_3 &= 0\end{aligned}$$

Thus letting  $\omega = \omega_3(0) (I - I_3)/I$  we have as a solution

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} \cos \omega t & \sin \omega t & 0 \\ -\sin \omega t & \cos \omega t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \omega_1(0) \\ \omega_2(0) \\ \omega_3(0) \end{bmatrix}$$

If we select the initial time in a suitable way then  $\omega_2(0) = 0$  and the kinematical equations are  $\underline{\dot{A}} = \underline{\Omega} \underline{A}$  with  $\underline{\Omega}$  given by

## INPUT DATA SHEET

### I. Kinematic Specifications

Fill in either A or B

#### A. Initial Values of Direction Cosines

$$\underline{A}(0) = \begin{pmatrix} \underline{1} & \underline{0} & \underline{0} \\ \underline{0} & \underline{\sqrt{3}/2} & \underline{1/2} \\ \underline{0} & \underline{-1/2} & \underline{\sqrt{3}/2} \end{pmatrix}$$

#### B. Initial Euler Angles

$$\theta_o = \underline{\hspace{2cm}} \text{ rad}, \varphi_o = \underline{\hspace{2cm}} \text{ rad}, \psi_o = \underline{\hspace{2cm}} \text{ rad}$$

### II. Dynamic Specifications

The  $i^{\text{th}}$  moment of inertia and the  $i^{\text{th}}$  component of angular velocity are assumed to be taken about the  $i^{\text{th}}$  principal axis of the satellite. If there is a reaction wheel, the axes must be numbered so as to place the wheel on the 3<sup>rd</sup> axis.

#### A. Satellite Characteristics

Initial value of angular velocity:

$$\underline{\omega}(0) = \begin{pmatrix} \underline{0 \frac{\text{rad}}{\text{sec}}} & \underline{\frac{6\sqrt{3}}{15} \frac{\text{rad}}{\text{sec}}} & \underline{1 \frac{\text{rad}}{\text{sec}}} \end{pmatrix}$$

Moments of inertia:

$$I_1 = \underline{5 \text{ kg m}^2}, I_2 = \underline{5 \text{ kg m}^2}, I_3 = \underline{6 \text{ kg m}^2}$$

#### B. Rotor Characteristics

If there is a spherical damping rotor on the  $i^{\text{th}}$  axis, fill in  $\chi_i(0)$ , its initial angular velocity;  $R_i$ , its moment of inertia; and  $k_i$ , its damping constant. If no rotor, write NONE in the corresponding blanks.

$$\begin{aligned} \underline{\chi}(0) &= \begin{pmatrix} \underline{\text{NONE} \frac{\text{rad}}{\text{sec}}} & \underline{\text{NONE} \frac{\text{rad}}{\text{sec}}} & \underline{\text{NONE} \frac{\text{rad}}{\text{sec}}} \end{pmatrix} \\ R_1 &= \underline{\text{NONE} \text{ kg m}^2}, R_2 = \underline{\text{NONE} \text{ kg m}^2}, R_3 = \underline{\text{NONE} \text{ kg m}^2} \\ k_1 &= \underline{\text{NONE} \frac{\text{kg m}^2}{\text{sec}}}, k_2 = \underline{\text{NONE} \frac{\text{kg m}^2}{\text{sec}}}, k_3 = \underline{\text{NONE} \frac{\text{kg m}^2}{\text{sec}}} \end{aligned}$$



### C. Reaction Wheel Characteristics

If the satellite has a reaction wheel, fill in  $\chi_3$ , its initial angular velocity, and  $W_1$ ,  $W_2$ ,  $W_3$ , its moments of inertia. Also supply a control law determining  $\ell_3(\underline{A}, \underline{\omega}, \chi_3)$ , the torque of the reaction wheel on the satellite. If no wheel, write NONE.

$$\begin{aligned}\chi_3 &= \underline{\text{NONE}} \frac{\text{rad}}{\text{sec}} \\ W_1 = W_2 &= \underline{\text{NONE}} \text{ kg m}^2, & W_3 &= \underline{\text{NONE}} \text{ kg m}^2 \\ \ell_3 &= \underline{\text{NONE}}\end{aligned}$$

### III. External Torque

Supply a control law determining  $\underline{n}$ , the external torque on the satellite, in body-fixed coordinates.

$$\begin{aligned}n_1 &= \underline{\hspace{10em}0\hspace{10em}} \text{ nt m} \\ n_2 &= \underline{\hspace{10em}0\hspace{10em}} \text{ nt m} \\ n_3 &= \underline{\hspace{10em}0\hspace{10em}} \text{ nt m}\end{aligned}$$

### IV. Termination Criterion

Fill in A or B, or both. If both, the first criterion to be met will terminate the simulation.

- A. Terminate after integrating over a period of 10,800 sec.  
B. Run the simulation until the solutions satisfy the following condition:

---

---

---

3-63-0029

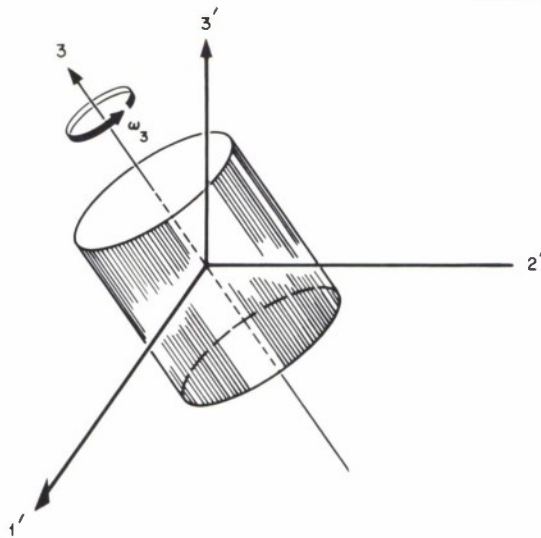


Fig. 4. Sample problem Number 1. Torque free motion  $I_1 = I_2 = 5$ ;  $I_3 = 6$ .

$$\Omega(t) = \begin{bmatrix} 0 & \omega_3(0) & -\omega_1(0) \sin \omega t \\ -\omega_3(0) & 0 & \omega_1(0) \cos \omega t \\ \omega_1(0) \sin \omega t & -\omega_1(0) \cos \omega t & 0 \end{bmatrix}$$

as can be readily inferred from the classical treatments

$$\underline{\underline{A}}(t) = \begin{bmatrix} \cos \omega t & \sin \omega t & 0 \\ -\sin \omega t & \cos \omega t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \dot{\phi}_0 t & \sin \dot{\phi}_0 t & 0 \\ -\sin \dot{\phi}_0 t & \cos \dot{\phi}_0 t & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

where

$$\cos \theta = I_3 \omega_3 / \left( \sqrt{I_1^2 \omega_1^2 + I_2^2 \omega_2^2 + I_3^2 \omega_3^2} \right)$$

$$\dot{\phi}_0 = \frac{\sqrt{\omega_1^2 + \omega_2^2}}{\sin \theta}$$

In the case of torque-free motion, the Adams-Moulton solution curves can be compared directly with the exact solutions. Let

$$\eta_{ij}(t) = |(a_{ij}(t) - \hat{a}_{ij}(t)) / \hat{a}_{ij}(t)|$$

where  $a_{ij}(t)$  is the  $ij^{\text{th}}$  element of  $\underline{\underline{A}}(t)$  as computed by Adams-Moulton integration, and  $\hat{a}_{ij}(t)$  is the corresponding element as found by evaluating the expression for the exact solution. Then calculate the mean value

$$\mu_A(t) = \frac{1}{9} \sum_{i,j} \eta_{ij}(t)$$

and the variance

$$\sigma_A^2(t) = \frac{1}{9} \sum_{i,j} (\eta_{ij}(t) - \mu_A(t))^2$$

Finally, let  $\eta_{\ell_{13}}(t)$  be the error in  $\ell_{13}(\underline{A}, t)$ , a component of angular momentum computed from the integral curves for  $\underline{A}$ . Since the angular momentum is constant in torque-free motion

$$\eta_{\ell_{13}}(t) = |(\ell_{13}(\underline{A}, t) - \ell_{13}(\underline{A}, 0)) / \ell_{13}(\underline{A}, 0)|$$

The values of  $\mu_A$ ,  $\sigma_A^2$ , and  $\eta_{\ell_{13}}$  tabulated below were calculated after integrating with an initial step size of 0.08 sec and with the truncation error in each function confined to the interval  $[10^{-7}, 8 \times 10^{-5}]$ . The ratio of simulated time to real time was 10.3 to 1.

t	$\mu_A(t)$	$\sigma_A^2(t)$	$\eta_{\ell_{13}}(A)$
3600 sec	0.0012	$2.0 \times 10^{-6}$	0.0011
7200 sec	0.0021	$4.4 \times 10^{-6}$	0.0022
10800 sec	0.016	$1.0 \times 10^{-3}$	0.0033

The anomalous value of  $\mu_A(10,800)$  is more likely an indication of high error in the calculated "exact" solution than of high error in the integral curves. This conclusion is supported by the high value of  $\sigma_A^2(10,800)$  and the low value of  $\eta_{\ell_{13}}(10,800)$ . The preceding values of  $\mu_A$  may also be considerably higher than the actual integration errors. In fine, with the initial step size and error bounds given above, and with an integration period of three hours, the mean error in the nine direction cosine solution functions increases, in all likelihood, by no more than 0.1% per hour.

Since the expressions for the exact solutions for  $\underline{w}$  do not contain products of trigonometric functions of  $t$ , the error in these functions is not compounded. Moreover, the integrated values of  $\underline{w}$  were in perfect agreement with the exact solutions; during the entire three hours of simulated time, there was no detectable error in  $\underline{w}$ . Since the integrated and the calculated solutions were known to the nearest  $.00001 \text{ sec}^{-1}$ , the maximum absolute error in  $\underline{w}$  was  $5 \times 10^{-6} \text{ sec}^{-1}$ .

## VII. SAMPLE PROBLEM 2: A MAGNETIC ATTITUDE CONTROL LAW

We now illustrate how the program might be used to check out an attitude control system. We assume a spinning cylindrical body with two damping rotors and no reaction wheel. We also assume magnetic torquing rods perpendicular to each other and the spin axis. For simplicity we examine a fixed

magnetic field which is parallel with the desired orientation of the spin axis. The dynamical equations, a synthesis of the basic equations of Sections II, III and IV are

$$(I + 2R) \dot{\omega}_1 = (I - I_3) \omega_3 \omega_2 - k(\omega_1 - \chi_1) + n_1$$

$$(I + 2R) \dot{\omega}_2 = (I_3 - I) \omega_3 \omega_1 - k(\omega_2 - \chi_2) + n_2$$

$$(2R + I_3) \dot{\omega}_3 = n_3$$

$$R \dot{x}_1 = k(\omega_1 - \chi_1)$$

$$R \dot{x}_2 = k(\omega_2 - \chi_2)$$

where

$$n_1 = u_2 a_{33}$$

$$n_2 = -u_1 a_{33}$$

$$n_3 = u_1 a_{23} - u_2 a_{13}$$

and the u's (proportional to the strength of the respective magnets) are given by the control law

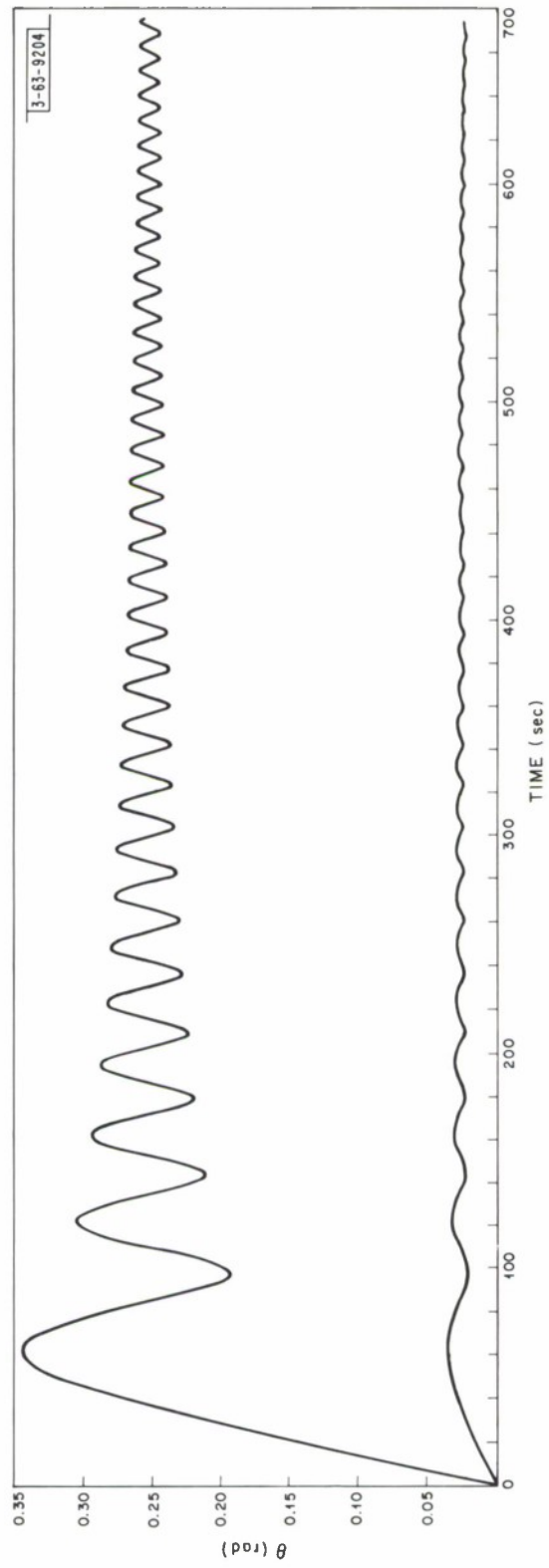
$$u_1 = u_0 \operatorname{sgn} [a_{33} (a_{22} \ell_2 - a_{12} \ell_{23})]$$

$$u_2 = u_0 \operatorname{sgn} [a_{33} (a_{11} \ell_{23} - a_{12} \ell_{31})]$$

That is, the u's are chosen in such a way as to drive the angular momentum vector to the vertical. The numerical constants can be found on the input data sheets.

A plot of the angle between the vertical and the  $\omega_3$  axis is shown for two particular values of  $u_0$ .





## INPUT DATA SHEET

### I. Kinematic Specifications

Fill in either A or B

#### A. Initial Values of Direction Cosines

$$\underline{A}(0) = \begin{pmatrix} \underline{1} & \underline{0} & \underline{0} \\ \underline{0} & \underline{\sqrt{3}/2} & \underline{1/2} \\ \underline{0} & \underline{-1/2} & \underline{\sqrt{3}/2} \end{pmatrix}$$

#### B. Initial Euler angles

$$\theta_0 = \underline{\hspace{2cm}} \text{ rad}, \varphi_0 = \underline{\hspace{2cm}} \text{ rad}, \psi_0 = \underline{\hspace{2cm}} \text{ rad}$$

### II. Dynamic Specifications

The  $i^{\text{th}}$  moment of inertia and the  $i^{\text{th}}$  component of angular velocity are assumed to be taken about the  $i^{\text{th}}$  principal axis of the satellite. If there is a reaction wheel, the axes must be numbered so as to place the wheel on the 3rd axis.

#### A. Satellite Characteristics

Initial Value of Angular Velocity

$$\underline{\omega}(0) = \left( \underline{0} \frac{\text{rad}}{\text{sec}}, \underline{0} \frac{\text{rad}}{\text{sec}}, \underline{5} \frac{\text{rad}}{\text{sec}} \right)$$

Moments of Inertia:

$$I_1 = \underline{5} \text{ kg m}^2, I_2 = \underline{5} \text{ kg m}^2, I_3 = \underline{6} \text{ kg m}^2$$

#### B. Rotor Characteristics

If there is a spherical damping rotor on the  $i^{\text{th}}$  axis, fill in  $\chi_i(0)$ , its initial angular velocity;  $R_i$ , its moment of inertia; and  $k_i$ , its damping constant. If no rotor, write NONE in the corresponding blanks.

$$\underline{\chi}(0) = \left( \underline{0} \frac{\text{rad}}{\text{sec}}, \underline{0} \frac{\text{rad}}{\text{sec}}, \underline{\text{NONE}} \frac{\text{rad}}{\text{sec}} \right)$$

$$R_1 = \underline{.05} \text{ kg m}^2, R_2 = \underline{.05} \text{ kg m}^2, R_3 = \underline{\text{NONE}} \text{ kg m}^2$$

$$k_1 = \underline{.01} \frac{\text{kg m}^2}{\text{sec}}, k_2 = \underline{.01} \frac{\text{kg m}^2}{\text{sec}}, k_3 = \underline{\text{NONE}} \frac{\text{kg m}^2}{\text{sec}}$$

### C. Reaction Wheel Characteristics

If the satellite has a reaction wheel, fill in  $\chi_3$ , its initial angular velocity, and  $W_1$ ,  $W_2$ ,  $W_3$ , its moments of inertia. Also supply a control law determining  $\ell_3$  ( $\underline{A}$ ,  $\underline{\omega}$ ,  $\chi_3$ ), the torque of the reaction wheel on the satellite. If no wheel, write NONE.

$$\begin{aligned}\chi_3 &= \underline{\text{NONE}} \quad \frac{\text{rad}}{\text{sec}} \\ W_1 = W_2 &= \underline{\text{NONE}} \quad \text{kg m}^2 & W_3 &= \underline{\text{NONE}} \quad \text{kg m}^2 \\ \ell_3 &= \underline{\text{NONE}}\end{aligned}$$

### III. External Torque

Supply a control law determining  $\underline{n}$ , the external torque on the satellite, in body-fixed coordinates. Let  $u_1 = u_0 \operatorname{sgn} (a_{22} \ell_2 - a_{12} \ell_1) a_{33}$  and  $u_2 = u_0 \operatorname{sgn} (a_{11} \ell_1 - a_{12} \ell_2) a_{33}$

$$\begin{aligned}n_1 &= \underline{u_2 a_{33}} \quad \text{nt m} \\ n_2 &= \underline{-u_1 a_{33}} \quad \text{nt m} \\ n_3 &= \underline{u_1 a_{23} - u_2 a_{13}} \quad \text{nt m}\end{aligned}$$

where  $u_0$  is either .01 or .002

### IV. Termination Criterion

Fill in A or B, or both. If both, the first criterion to be met will terminate the simulation.

- A. Terminate after integrating over a period of 1500 sec.
- B. Run the simulation until the solutions satisfy the following condition:

---



---



---

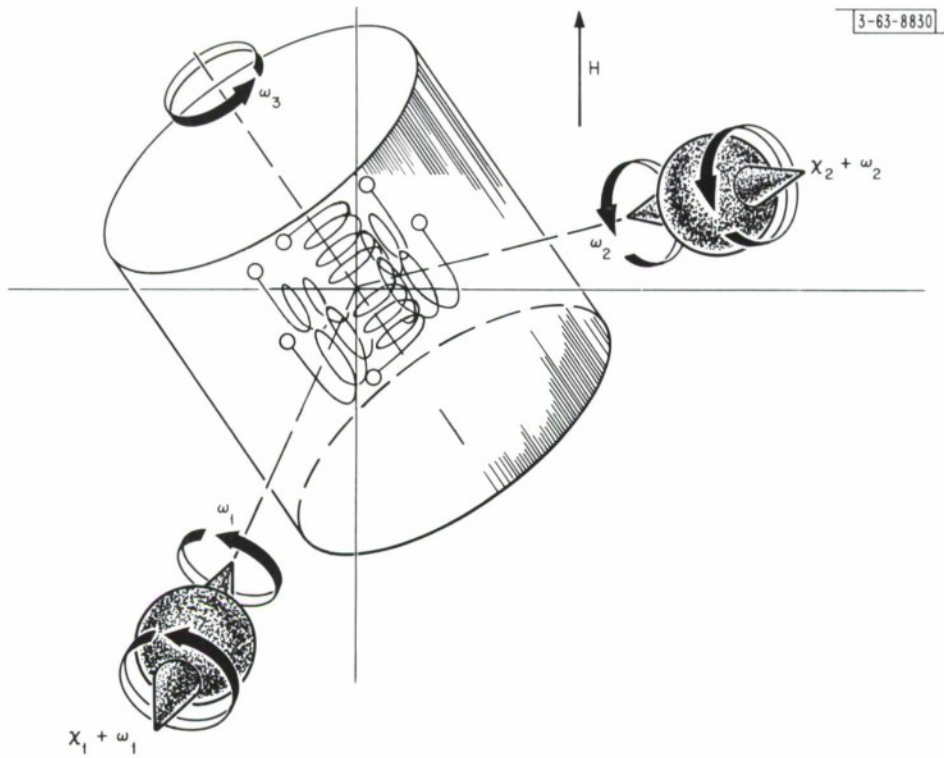


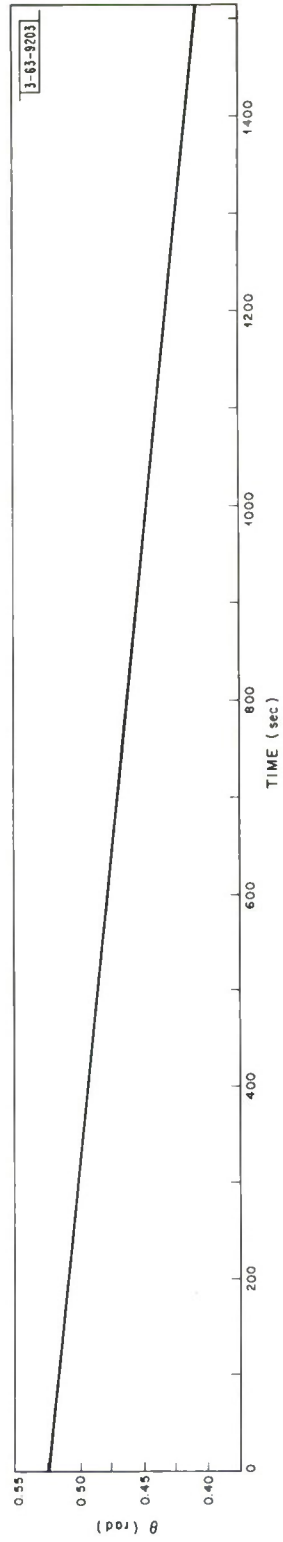
Fig. 5. Sample problem Number 2. Magnetic torquing with a fixed magnetic field. Damping rotors are included.

### VIII. SAMPLE PROBLEM 3: REACTION WHEEL SPIN DOWN

Consider a cylindrical body whose moment of inertia along the axis of symmetry is smaller than the moment of inertia along the other two principle axes. Such a body might be held relatively stable by putting a spinning wheel along the axis of symmetry with an electric motor drive to overcome friction. We want to investigate what happens if the power fails and the wheel spins down due to friction. The main item of interest might be the angle that the axis of symmetry makes with the vertical after a certain period of time. Naturally if the system is perfectly balanced and  $\omega_1 = \omega_2 = 0$  initially then momentum lost by the wheel will be taken up by the body but no change in the orientation of the spin axis will occur. However if the initial values of  $\omega_1$  and  $\omega_2$  are slightly different from zero then wobble will build up. We assume the existence of damping rotors on the 1- and 2-axes and take the reaction wheel along the 3-axis. Let the moments of inertia be  $I_1 = I_2 = 5$  and  $I_3 = 3$ . The initial velocity of the main body is zero and is assumed to be perfectly oriented ( $\underline{A}(0)$  is the identity matrix). Since the complete input data sheet is attached, we only remark that the initial angular momentum of the wheel is  $180 \text{ kg m}^2/\text{sec}$ . The value of damping picked is such that the wheel spin down has a time constant of 18 minutes. Attached graph illustrates the misorientations of the spin axis as a function of time.

When the reaction wheel spin down was simulated with an initial step size of 0.04 sec and with the same error bounds as in the torque-free motion simulation, the ratio of simulation time to real time was 11 to 1.





## INPUT DATA SHEET

### I. Kinematic Specifications

Fill in either A or B

A. Initial Values of Direction Cosines:

$$\underline{A}(0) = \begin{pmatrix} \underline{\quad 1 \quad} & \underline{\quad 0 \quad} & \underline{\quad 0 \quad} \\ \underline{\quad 0 \quad} & \underline{\quad 1 \quad} & \underline{\quad 0 \quad} \\ \underline{\quad 0 \quad} & \underline{\quad 0 \quad} & \underline{\quad 1 \quad} \end{pmatrix}$$

B. Initial Euler Angles:

$$\theta_0 = \underline{\hspace{2cm}} \text{ rad}, \varphi_0 = \underline{\hspace{2cm}} \text{ rad}, \psi_0 = \underline{\hspace{2cm}} \text{ rad}$$

### II. Dynamic Specifications

The  $i^{\text{th}}$  moment of inertia and the  $i^{\text{th}}$  component of angular velocity are assumed to be taken about the  $i^{\text{th}}$  principal axis of the satellite. If there is a reaction wheel, the axes must be numbered so as to place the wheel on the 3<sup>rd</sup> axis.

A. Satellite Characteristics

Initial value of angular velocity:

$$\underline{\omega}(0) = \begin{pmatrix} \underline{\quad .05 \quad \frac{\text{rad}}{\text{sec}} \quad}, & \underline{\quad .05 \quad \frac{\text{rad}}{\text{sec}} \quad}, & \underline{\quad 0 \quad \frac{\text{rad}}{\text{sec}} \quad} \end{pmatrix}$$

Moments of Inertia:

$$I_1 = \underline{\quad 5 \text{ kg m}^2 \quad}, \quad \underline{\quad 5 \text{ kg m}^2 \quad}, \quad \underline{\quad 3 \text{ kg m}^2 \quad}$$

B. Rotor Characteristics

If there is a spherical damping rotor on the  $i^{\text{th}}$  axis, fill in  $\chi_i(0)$ , its initial angular velocity;  $R_i$ , its moment of inertia; and  $k_i$ , its damping constant. If no rotor, write NONE in the corresponding blanks.

$$\underline{\chi}(0) = \begin{pmatrix} \underline{\quad 0 \quad \frac{\text{rad}}{\text{sec}} \quad}, & \underline{\quad 0 \quad \frac{\text{rad}}{\text{sec}} \quad}, & \underline{\quad \text{NONE} \quad \frac{\text{rad}}{\text{sec}} \quad} \end{pmatrix}$$

$$R_1 = \underline{\quad .05 \text{ kg m}^2 \quad}, R_2 = \underline{\quad .05 \text{ kg m}^2 \quad}, R_3 = \underline{\quad \text{NONE} \text{ kg m}^2 \quad}$$

$$k_1 = \underline{\quad .01 \frac{\text{kg m}^2}{\text{sec}} \quad}, k_2 = \underline{\quad .01 \frac{\text{kg m}^2}{\text{sec}} \quad}, k_3 = \underline{\quad \text{NONE} \frac{\text{kg m}^2}{\text{sec}} \quad}$$

### C. Reaction Wheel Characteristics

If the satellite has a reaction wheel, fill in  $\chi_3$ , its initial angular velocity, and  $W_1$ ,  $W_2$ ,  $W_3$ , its moments of inertia. Also supply a control law determining  $\ell_3(A, \omega, \chi_3)$ , the torque of the reaction wheel on the satellite. If no wheel write NONE.

$$\begin{aligned}\chi_3 &= \frac{60}{\text{sec}} \frac{\text{rad}}{\text{sec}} \\ W_1 = W_2 &= \frac{.1 \text{ kg m}^2}{(\chi_3 - \omega_3) \times 10^{-4}}, & W_3 &= \frac{.1 \text{ kg m}^2}{\ell_3} \\ \ell_3 &= \end{aligned}$$

### III. External Torque

Supply a control law determining  $\underline{n}$ , the external torque on the satellite, in body-fixed coordinates.

$$\begin{aligned}n_1 &= \frac{0}{\text{nt m}} \\ n_2 &= \frac{0}{\text{nt m}} \\ n_3 &= \frac{0}{\text{nt m}}\end{aligned}$$

### IV. Termination Criterion

Fill in A or B, or both. If both, the first criterion to be met will terminate the simulation.

A. Terminate after integrating over a period of 300 sec.

B. Run the simulation until the solutions satisfy the following condition:

---



---



---

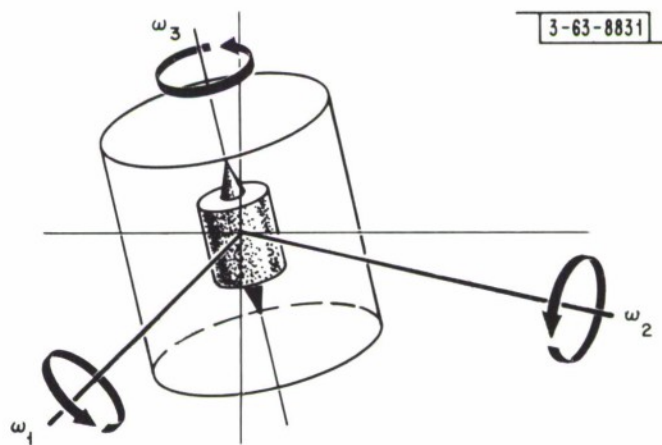


Fig. 6. Wheel spin down problem. Damping rotors on 1 and 2 axes; reaction wheel on 3 axis. Initially the rigid body has angular velocities  $\omega_1 = .05$   $\omega_2 = .05$   $\omega_3 = 0$ .

## APPENDIX 1

### USAGE OF THE ATTITUDE STABILIZATION PROGRAM

#### Purpose of the Program

The purpose of the attitude stabilization program is to simulate the motion of a satellite about its center of mass. The motion is described by a system of differential equations of the form:

$$\begin{aligned}
 \dot{a}_{11}(t) &= f_1(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{12}(t) &= f_2(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{13}(t) &= f_3(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{21}(t) &= f_4(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{22}(t) &= f_5(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{23}(t) &= f_6(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{31}(t) &= f_7(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{32}(t) &= f_8(t, \underline{\omega}, \underline{A}) \\
 \dot{a}_{33}(t) &= f_9(t, \underline{\omega}, \underline{A}) \\
 \dot{\omega}_1(t) &= f_{10}(t, \underline{\omega}, \underline{\chi}, \underline{A}) \\
 \dot{\omega}_2(t) &= f_{11}(t, \underline{\omega}, \underline{\chi}, \underline{A}) \\
 \dot{\omega}_3(t) &= f_{12}(t, \underline{\omega}, \underline{\chi}, \underline{A}) \\
 \dot{\chi}_1(t) &= f_{13}(t, \underline{\omega}, \underline{\chi}) \\
 \dot{\chi}_2(t) &= f_{14}(t, \underline{\omega}, \underline{\chi}) \\
 \dot{\chi}_3(t) &= f_{15}(t, \underline{\omega}, \underline{\chi})
 \end{aligned}$$

where  $\underline{\omega} = (\omega_1, \omega_2, \omega_3)$ ,  $\underline{\chi} = (\chi_1, \chi_2, \chi_3)$ , and

$$\underline{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$



More concisely, in vector notation, the system can be written:

$$\dot{\underline{y}}(t) = \underline{f}(t, \underline{y}), \text{ where } \underline{f} = (f_1, f_2, \dots, f_{15}), \underline{y} = (a_{11}, a_{12}, \dots, a_{33}, \omega_1, \dots, \chi_3), \text{ and } \dot{\underline{y}} = (\dot{a}_{11}, \dot{a}_{12}, \dots, \dot{a}_{33}, \dot{\omega}_1, \dots, \dot{\chi}_3).$$

The simulation is accomplished by using techniques of numerical integration to compute the solution curves of the above system:  $\underline{A}$ ,  $\underline{\omega}$ , and  $\underline{\chi}$ .

### Numerical Integration

Given  $\eta$ ,  $t_0$ , and the differential equation  $\dot{y}(t) = f(t, y)$ , there is one and only one function  $y$  such that  $dy/dt = \dot{y}$  and  $y(t_0) = \eta$ . By numerical integration, one can find a good approximation to  $y$ , the solution function, on a set of discrete points,  $\{t_i\}$ . The initial value,  $\eta$ , tells where  $y$  is located at a certain time,  $t_0$ , and the value of the derivative at this point,  $f(t_0, \eta)$ , indicates where  $y$  is headed. Therefore, if  $h$  is a small time interval, one can compute with little error where  $y$  will be at  $t_1 = t_0 + h$ . Then  $y(t_1)$  can be used to evaluate  $f(t_1, y(t_1))$ , which in turn will point the way to  $y$  at  $t_2 = t_1 + h$ . The process of getting from  $y(t_i)$  to  $y(t_{i+1})$  is called an integration step, and the step size is  $h = t_{i+1} - t_i$ .

A method of numerical integration is called self-starting if the only input data required is  $t_0$  and  $\eta$ . The Adams-Moulton predictor-corrector method, which is used for the attitude stabilization simulations, is not self-starting. Only one value of  $\dot{y}$  can be computed from  $t_0$  to  $\eta$ , but the Adams-Moulton formulas for  $y(t_{i+1})$  require values of  $\dot{y}$  at four consecutive points:  $t_{i-3}$ ,  $t_{i-2}$ ,  $\dots$ ,  $t_i$ . Therefore, the program includes a Runge-Kutta integration routine, which computes the starting values needed for the Adams-Moulton integration.

### Runge-Kutta Integration Routine

A Runge-Kutta integration step has the form  $y(t + h) = y(t) + h * \Phi(t)$ , where  $\Phi(t)$  is calculated as follows:

$$\begin{aligned}
G1(t) &= f(t, y) = \dot{y}(t) \\
Z1(t) &= y(t) + 0.5h * G1(t) \\
G2(t) &= f(t + 0.5h, Z1) \\
Z2(t) &= y(t) + 0.5h * G2(t) \\
G3(t) &= f(t + 0.5h, Z2) \\
Z3(t) &= y(t) + h * G3(t) \\
G4(t) &= f(t + h, Z3) \\
\Phi(t) &= (G1(t) + 2G2(t) + 2G3(t) + G4(t))/6
\end{aligned}$$

When a system of equations,  $\dot{\underline{y}}(t) = \underline{f}(t, \underline{y})$ , is being integrated, each function  $y_j$  in  $\underline{y}$  has its corresponding  $G1_j, Z1_j, \dots, G4_j, \Phi_j$ , which are computed exactly as above, using  $f_j$  to compute  $G1_j, \dots, G4_j$ . It is important to note that all components of the vector  $\underline{Z1}$  must be calculated before any of the components of  $\underline{G2}$ , all of  $\underline{Z2}$  before any of  $\underline{G3}$ , all of  $\underline{Z3}$  before any of  $\underline{G4}$ , and all of  $\underline{y}$  before any of the next  $\underline{G1}$ . These are the only restrictions on the order of computations.

In the  $\underline{\dot{A}}, \underline{\dot{\omega}}, \underline{\dot{\chi}}$  system,  $t$  appears implicitly, as in  $\underline{\omega}$ , but never explicitly, as in  $\sin t$ ,  $e^t$ , or  $3t + 10$ . Therefore, the Runge-Kutta equations reduce to a simpler form for this system:

$$\begin{aligned}
G1 &= f(y) \\
Z1 &= y + 0.5h * G1 \\
G2 &= f(Z1) \\
Z2 &= y + 0.5h * G2 \\
G3 &= f(Z2) \\
Z3 &= y + h * G3 \\
G4 &= f(Z3) \\
\Phi &= (G1 + 2G2 + 2G3 + G4)/6 \\
y &= y + h * \Phi
\end{aligned}$$

In the following summary of the Runge-Kutta routine, and in the summaries of every subroutine described below, all CALL's will be discussed, and

all arguments and COMMON variables will be listed, together with all type, length, and dimension specifications not implied by FORTRAN convention; since no IMPLICIT statements appear in the program, no nonstandard conventions pertain. None of the COMMON blocks is used for subroutine-to-subroutine communication; they are used only for initializing subroutine variables to values set in the main program, WCHA. Variables labeled "input" contain data supplied to the subroutine in question; those labeled "output" contain data computed by subroutine.

The Runge-Kutta subroutine, RK(H, T, Y, FY), computes the starting values for Adams-Moulton.

#### A. Arguments

H, REAL \* 8; input; step size

T, REAL \* 8; as input,  $T = t_{i-3}$ ; as output,  $T = t_i$ .

Y(15), REAL \* 8; as input,  $Y = \underline{y}(t_{i-3})$ ; as output,

$Y = \underline{y}(t_i)$ . The functions are ordered as above, in the description of  $\underline{A}$ ,  $\underline{\omega}$ ,  $\underline{\chi}$  system.

FY(15, 5), REAL \* 8; output;  $FY(J, K) = f_J(t_{i-(4-K)}, \underline{Y})$ ,

$J = 1, 15$ ,  $K = 1, 3$ , where  $f_J$  is the derivative of

$Y_J$ .  $FY(J, K) = 0$ ,  $J = 1, 15$ ,  $K = 4, 5$ .

#### B. COMMON

/CHIDOT/YDOT(15)

REAL \* 8: YDOT

### Adams-Moulton Integration Routine

For the differential equation  $\dot{y}(t) = f(t, y)$ , an Adams-Moulton integration step has the form:

```

y(t + h) = y(t) + h * p(t, f)
f_p      = f(t + h, y(t + h))
DO 1 m = 1, n
y(t + h) = y(t) + h * c(t, f, f_p)
1  f_p    = f(t + h, y(t + h))

```

where the functions  $p$  and  $c$  are given by the predictor and corrector formulas described below.

Increasing  $n$ , the number of iterations of the corrector formula, has an effect and a side effect: the error decreases but the computation time increases. On account of the gravity of the side effect,  $n = 1$  in the Adams-Moulton subroutine; the first corrected value is taken as  $y(t + h)$ .

For  $g$ , an arbitrary function of time, let  $g_n = g$  at  $t_n$ ,  $n = i - 3, i - 2, \dots, i + 1$ . Then the predictor formula can be written

$$y_{i+1} = y_i + (55 f_i - 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3}) * h/24$$

The predicted value of  $y_{i+1}$  is used to calculate  $f_{i+1}$ , which appears in the corrector formula:

$$y_{i+1} = y_i + (9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2}) * h/24$$

To apply the Adams-Moulton method to a system of equations,  $\dot{\underline{y}} = \underline{f}(t, \underline{y})$ , simply apply the predictor and corrector formulas to each component of  $\underline{y}$ , making sure to compute the entire predicted vector  $\underline{y}_{i+1}$  before starting to calculate  $\underline{f}_{i+1}$  for the corrector formula.

It is important to maintain an optimum step size during the integration. If  $h$  is too small, the computation takes too much time, but if  $h$  is too big, there is a large error in the solutions. To optimize  $h$ , an error estimate,  $E$ , is calculated for each of the fifteen functions after every integration step.  $E_{\max}$ , the greatest of the fifteen errors, has upper and lower bounds specified as input to the program. If  $E_{\max}$  is greater than LUB, the upper bound, then  $h$  will be halved, and if  $E_{\max}$  is less than GLB, the lower bound, then  $h$  will be doubled.

Whenever the step size is changed, the Runge-Kutta starting routine must be called, for the Adams-Moulton routine cannot find the complete set of derivatives at the new points  $\{t_{i-3}, t_{i-2}, t_{i-1}\} = \{t_i - 3h, t_i - 2h, t_i - h\}$  unless the step size has remained constant.

The choice of the lower and upper error bounds, GLB and LUB, will be a compromise between accuracy and speed. Often a good set of bounds is  $GLB = 10^{-7}$ ,  $LUB = 10^{-5}$ . However, an optimum set of bounds for a particular  $\dot{A}$ ,  $\dot{\omega}$ ,  $\dot{\chi}$  system can be determined as follows. Form  $n$  sets of bounds,  $B_1, \dots, B_n$ ;  $B_i = \{GLB, LUB\}_i$ ,  $i = 1, \dots, n$ . Run  $n$  simulations, identical except in the choice of  $B_i$ , and obtain  $n$  corresponding sets of solutions,  $S_1, \dots, S_n$ ;  $S_i = \{\underline{A}, \underline{\omega}, \underline{\chi}\}_i$ ,  $i = 1, n$ . Then study the discrepancy between  $S_i$  and  $S_j$ , for  $\{i, j : i \neq j\}$ . Suppose  $B_i$  has more restrictive bounds than  $B_j$ ; i.e.,  $h$  will halve more easily and/or double less easily under  $B_i$ . Then if the discrepancy between  $S_i$  and  $S_j$  is great, one may reject  $B_j$  as unacceptably lax. If, on the other hand, the discrepancy is slight, and if  $S_j$  took substantially less execution time than  $S_i$ , then  $B_j$  is to be preferred to  $B_i$ , for it provides greater speed with comparable accuracy.

In order to compare  $S_i$  with  $S_j$ , locate  $m$  points of time,  $t_1, t_2, \dots, t_m$ , at which solutions appear in both sets of output. Assuming  $B_i$  more restrictive than  $B_j$ , a good measure of the discrepancy at  $t = t_k$  is given by:

$$\overline{D}_k = \frac{1}{15} \sum_{n=1}^{15} | (s_{in}(t_k) - s_{jn}(t_k)) / s_{in}(t_k) |$$

where  $s_{in}$  and  $s_{jn}$  are the  $n^{th}$  functions in  $S_i$  and  $S_j$  respectively. If  $\overline{D}_k$  is small for  $k = 1, \dots, m$ , then  $B_i$  and  $B_j$  provide about the same accuracy.

Be sure, when seeking to optimize GLB and LUB, that the initial value of the step size,  $h_0$ , is reasonable. If  $h_0$  is too big, the accuracy will be poor no matter how small an upper bound is used. An initial step size of 0.05 sec is usually satisfactory for the WCHA program. However, as a precaution, it might be wise to begin with  $h_0 = 0.001$  sec,  $GLB = 10^{-7}$ ,  $LUB = 10^{-5}$ , and see how the step size behaves over an integration period of, say, 60 sec. Having thus determined what range of step sizes is appropriate for the system at hand, select  $h_0$  from the lower end of that range.



## Summary

The Adams-Moulton subroutine, AM(H, T, Y, FY, GLB, LUB), computes the integral curves for the system  $\dot{\underline{A}}, \dot{\underline{\omega}}, \dot{\underline{\chi}}$ .

### A. Arguments

H, REAL \* 8; input; initial step size.

T, REAL \* 8; input;  $T = t_i$

Y(15), REAL \* 8; input;  $Y = \underline{y}(t_i)$

FY(15, 5), REAL \* 8; input;  $FY(J, K) = f_J(t_i - (4 - K)H, \underline{Y})$ ,  $J = 1, 15$ ,  
 $K = 1, 3$ . For  $K = 4, 5$ , FY is computed within the subroutine.

GLB; input; greatest lower bound on error.

LUB, REAL; input; least upper bound on error.

### B. Common

/CHIDOT/YDOT(15);/COEF/P(4), C(4)

REAL \* 8: YDOT, P, C

### C. Subroutines Called

DERIV; evaluates the derivatives.

OUTPUT; writes out the solutions, and determines when to terminate the simulation.

RK; restarts AM when step size changes.

## Auxiliary Subroutines

### I. ANGMOM

Subroutine ANGMOM (Y, EL) computes  $\underline{L}$ , the angular momentum of the satellite.  $\underline{L}$  is sometimes needed for the calculation of  $\underline{n}$  in subroutine TORQUE. In the special case of torque-free motion ( $\underline{n} \equiv \underline{0}$ ,  $\underline{\chi} \equiv \underline{0}$ ), the angular momentum should be perfectly constant; the accuracy of the integration can then be checked by writing out  $\underline{L}$  and observing its variance.

### A. Arguments

Y(15), REAL \* 8; input;  $Y = \underline{y}(t_i)$ .

EL(3, 3), REAL \* 8; output;  $EL(i, j) = L_{ij}$ ,  $i = 1, 3$ ,  $j = 1, 3$ .

B. Common

/MTM/OMEGA (3, 3), IINV (3, 3), DETI

REAL \* 8: OMEGA, IINV, DETI

II. DERIV

Subroutine DERIV (Y, YDOT) evaluates the derivative functions.

A. Arguments

Y(15), REAL \* 8; input; when the calling program is AM, either  $Y = \underline{y}_i$  [or  $\underline{y}(t_i)$ ] or  $Y = \underline{y}_{i+1}$  [or  $\underline{y}(t_{i+1})$ ]; when the calling program is RK, Y is either  $\underline{y}_i$  or one of the Runge-Kutta variables  $\underline{Z1}(t_i)$ ,  $\underline{Z2}(t_i)$ , or  $\underline{Z3}(t_i)$

YDOT(15), REAL \* 8; output;  $YDOT = \underline{dy}/dt$ , where  $\underline{y}$  is the input vector.

B. Common

/INIT/EN(3); /LOGIC/CH(3), TRQ, WHL; /PARAM/IA(3), R(3),  
KAY(3), W, W3.

LOGICAL: CH, TRQ, WHL

REAL \* 8: EN, IA, R, KAY, W, W3.

C. Subroutines Called

RCTN; evaluates the torque of the reaction wheel on the satellite.  
RCTN is called only if WHL = .TRUE..

TORQUE; evaluates the external torque on the satellite. TORQUE  
is called only if TRQ = .TRUE..

III. OUTPUT

Subroutine OUTPUT(T, Y, HLV, DBL, NSTEP, TERM) is a user-supplied output routine, called by AM after every integration step.

A. Arguments

T, REAL \* 8; input;  $T = t_i$

Y(15), REAL \* 8; input;  $Y = \underline{y}(t_i)$ .

HLV, INTEGER; input; number of times h has been halved.  
 DBL, INTEGER; input; number of times h has been doubled.  
 NSTEP; input; number of Adams-Moulton integration  
       steps which have been taken.  
 TERM, LOGICAL; output; setting TERM to .TRUE. will terminate  
       the simulation.

B. Common

/OTPT/WR, EPS, TBOUND

REAL \* 8: WR

REAL \* 4: EPS

/OTPT/ appears on WCHA for use in OUTPUT. TBOUND is an  
       upper limit on T; i. e. (T.GE.TBOUND) can be used as a  
       termination criterion. WR and EPS may be used to obtain  
       output every WR sec of simulation time. The following in-  
       structions will cause the solutions to be written out if and  
       only if  $|T - n * WR| \leq EPS$ ,  $n = 0, 1, 2, \dots$

TMWR = DMOD (T, WR)

IF (TMWR.GT.EPS.AND.(WR-TMWR).GT.EPS) GO TO 10

WRITE (6,20) T,Y

20 FORMAT ( . . . )

10 CONTINUE

Select an EPS small enough to constrain the output, yet not so small as to  
 prohibit it.

In choosing EPS, it is helpful to know the range and average size of h.  
 To find the average size, write out NSTEP, HLV, and DBL at the end of  
 simulation, and divide the period of integration by the total number, N, of  
 steps taken; N is given by  $NSTEP + 3*(HLV + DBL + 1)$ . To get an idea of  
 the range of h, make a trial run with a small TBOUND, a small WR, and a  
 large EPS. Another way to study the behavior of h is to write out HLV and  
 DBL periodically during the integration.

#### IV. MXPROD

Subroutine MXPROD (L, M, N, MXL, MXR, MXP) calculates matrix products.  $MXP = (MXL)(MXR)$ , where MXL is L by M, MXR is M by N, and MXP is L by N. All the matrices are REAL \* 8 arrays.

#### V. RCTN

Subroutine RCTN (Y, REAC) is a user-supplied routine which calculates the torque of the reaction wheel on the satellite. The form of the routine depends on the control law governing the reaction wheel.

##### A. Arguments

Y(15), REAL \* 8; input;  $Y = \underline{Y}(t_i)$

REAC, REAL \* 8; output; reaction torque on the satellite

#### VI. TORQUE

Subroutine TORQUE (Y, EN) is a user-supplied routine which calculates  $\underline{n}$ , the external torque in body-fixed coordinates. The form of the routine is determined by the control law governing the external torque.

##### A. Arguments

Y(15), REAL \* 8; input;  $Y = \underline{Y}(t_i)$ .

EN(3), REAL \* 8; output;  $EN = \underline{n}(t_i)$ .

#### Main Program

In the main program, WCHA, all the input data is read in, and all the COMMON variables are initialized. The input loop has the form:

```
1  READ(5, DATA)
   IF (HØ. EQ. Ø.) GO TO 6
   READ(5, 7) TITLE

7  FORMAT (20A4)
   [ Integrate.]
   GO TO 1

6  CONTINUE
   STOP
```

#### A. NAMELIST/DATA/

All the variables (except TITLE) are read in on a single list, /DATA/.  
The NAMELIST variables are:

H $\emptyset$ , REAL \* 8; initial step size. Last data card should set H $\emptyset$  to zero.

T $\emptyset$ , REAL \* 8; initial value of T, not destroyed.

Y $\emptyset$ (15), REAL \* 8; initial value of Y, not destroyed.

GLB; greatest lower bound on error.

LUB, REAL; least upper bound on error.

WR, REAL \* 8, and EPS(REAL \* 4); output can be obtained at desired intervals by writing out only when  $|T - n * WR| \leq EPS$ , for  $n = 0, 1, 2, \dots$

TBOUND; upper bound on T. TBOUND - T $\emptyset$  = period of integration.

TRQ, LOGICAL; TRQ = .TRUE. if there is a non-zero external torque, n, on the satellite. If n  $\equiv$  0, set TRQ to .FALSE..

IA(3), REAL \* 8; IA(j) = I<sub>j</sub>, the moment of inertia of the satellite about its j<sup>th</sup> principal axis.

R(3), REAL \* 8; if there is a spherical damping rotor on the j<sup>th</sup> axis, R(j) = R<sub>j</sub>, its moment of inertia. If there is no rotor on the j<sup>th</sup> axis, set R(j) to zero.

KAY(3), REAL \* 8; if there is a spherical damping rotor on the j<sup>th</sup> axis, KAY(j) = k<sub>j</sub>, its damping constant. If there is no rotor on the j<sup>th</sup> axis, set KAY(j) to zero.

W, W3, REAL \* 8; if the satellite has a reaction wheel, with moments of inertia W<sub>1</sub>, W<sub>2</sub>, W<sub>3</sub>, then W = W<sub>1</sub> = W<sub>2</sub>, and W3 = W<sub>3</sub>. If there is no reaction wheel, set W and W3 to zero.

#### B. TITLE

A title of up to 128 characters is printed at the head of each set of output. It is read in as an alphameric array TITLE(32). Since one data card contains only 80 characters, two cards must always be supplied for the title. If only one card is supplied, TITLE(21), TITLE(22), ..., TITLE(32) will be read from the following NAMELIST card, with dire consequences.

### C. COMMON

The COMMON blocks are listed below with the subroutines in which they appear.

```
/CHIDOT/YDOT (15); RK, AM  
/COEF/P (4), C(4); AM  
/INIT/EN (3); DERIV  
/LOGIC/CH (3), TRQ, WHL; DERIV  
/MTM/OMEGA (3, 3), IINV (3, 3), DETI; ANGMOM  
/OTPT/WR, EPS, TBOUND; OUTPUT  
/PARAM/IA (3), R (3), KAY (3), W, W3; DERIV  
LOGICAL: CH, TRQ, WHL  
REAL * 8: YDOT, P, C, EN, OMEGA, IINV, DETI, WR, IA, R, KAY, W, W3
```

### Summary

To use the attitude stabilization program, read in the input data as described above, and supply subroutines TORQUE, WHEEL, and OUTPUT. If TRQ = .FALSE., TORQUE will not be called, and if W = 0, WHEEL will not be called.

Before making any modifications in the program, study the interdependence of the subroutines involved, and double-check the specifications of the variables. The pertinent information is tabulated in the above subroutine profiles.



## REFERENCES

1. H. Goldstein, Classical Mechanics, Addison-Wesley, Reading, Mass. (1959).
2. S. McCuskey, An Introduction of Advanced Dynamics, Addison-Wesley, Reading, Mass. (1959).
3. E. Leimanis, The General Problem of the Motion of Coupled Rigid Bodies About a Fixed Point
4. P. Henrici, Discrete Variable Methods, John Wiley, New York (1962).
5. Guido Coupe, Magnetic Attitude Control of Satellites with Damping Rotors, S. M. Thesis, Dept. of Electrical Eng. M. I. T. (July 1966).
6. R. W. Brockett and B. J. Moriarty, "Magnetic Attitude Control of Spinning Satellites," JACC AIAA (preprints), New York (1966).

DOCUMENT CONTROL DATA - R&D		
<i>(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)</i>		
1. ORIGINATING ACTIVITY (Corporate author)  Lincoln Laboratory, M.I.T.		2a. REPORT SECURITY CLASSIFICATION Unclassified
		2b. GROUP None
3. REPORT TITLE  Numerical Simulation of the Attitude Equations of a Satellite Containing Damping Rotors and a Reaction Wheel		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Technical Note		
5. AUTHOR(S) (Last name, first name, initial)  Brockett, Roger W. Brody, Naomi M.		
6. REPORT DATE 15 November 1968	7a. TOTAL NO. OF PAGES 52	7b. NO. OF REFS 6
8a. CONTRACT OR GRANT NO. AF 19(628)-5167	9a. ORIGINATOR'S REPORT NUMBER(S) TN 1968-24	
b. PROJECT NO. 649L		
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) ESD-TR-68-352	
d.		
10. AVAILABILITY/LIMITATION NOTICES  This document has been approved for public release and sale; its distribution is unlimited.		
11. SUPPLEMENTARY NOTES  None	12. SPONSORING MILITARY ACTIVITY  Air Force Systems Command, USAF	
13. ABSTRACT  <p>This report describes a program for simulating the motion of a rigid body containing a reaction wheel, damping media and torquing (magnetic or otherwise). The dynamical equations are simulated in terms of the angular velocities in body-fixed coordinates and the kinematic equations are simulated in terms of direction cosines. The entire system of equations has been programmed using a fourth order predictor-corrector method with automatic step-size adjustment and a fourth order Runge-Kutta starting routine. The ratio of computer time to real time is between 1 to 10 and 2 to 1 for LES-5-type satellites. The potential user need only be familiar enough with this report to fill out the input data sheets on pages vii-ix.</p>		
14. KEY WORDS  satellite programmed equation system reaction wheel damping rotors  simulation of attitude equations program for simulating motion satellite motion		